DISTRIBUTED ENSEMBLE MODEL FOR PATTERN CLASSIFICATION OF IMBALANCED DATA

Synopsis of the Ph.D. Thesis

Submitted by

Varsha Sachin Khandekar

(Roll No. 70821016010)

Department of Computer Engineering

Supervisor

Dr. Pravin Shrinath

submitted in partial fulfillment of the requirements of

DOCTOR OF PHILOSOPHY (Engineering)

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING

SVKM's NARSEE MONJEE INSTITUTE OF MANAGEMENT STUDIES

(Declared as Deemed-to-be University Under Section 3 of the UGC Act, 1956)

V. L. Mehta Road, Vile Parle (West)

MUMBAI –400056

April, 2024

TABLE OF CONTENTS

SECTION NO.	TITLE	PAGE NO.
1	ABSTRACT	1
2	INTRODUCTION 2.1 Background 2.2 Research Gap	2
3	OBJECTIVES (key points, only one page)	5
4	LITERATURE REVIEW 4.1 Introduction 4.2 Binary and Multiclass Imbalanced Problems 4.3 Concept drift and Imbalanced issue in Data stream 4.4 Apache Spark	6
5	CONTRIBUTION 5.1 Objective I result and discussion 5.2 Objective II, results and discussion 5.3 Objective III results and discussion	11
6	CONCLUSION	28
7	FUTURE SCOPE	30
8	PUBLICATIONS	31
9	REFERENCES	32

ABSTRACT

Advancements in digital technology leads to explosion of data. Eventually to deal with this huge amount of data there is advancement in high performance computing, data analytic and machine learning. However, in real-world applications this data is not always in desired form and easy to analyse. There may be numerous complications in the data like noisy data, imbalanced underlying distribution of all the classes, data may be overlapping in nature. These problems become more complex in multi-class datasets as compared to binary class datasets. Also, most of the traditional machine learning algorithms were designed for binary class classification problems and can't be easily adapted as it is on multi-class classification problems or in distributed and parallel environment.

This research mainly focuses on multi-class classification problems in static as well as stream data. In the first part of our work, we investigated the applicability of existing resampling techniques (Random Undersampling, Random Over Sampling, SMOTE, ADASYN, Borderline-SMOTE, KMeans-SMOTE) in combination with bagging ensemble learning in a distributed environment. In this research we have implement enhanced K-Means-ASMOTE resampling technique using Spark. Proposed algorithm is tested on highly imbalanced datasets using Random Forest as ensemble classifier. Experimental results indicate that proposed model outperformed than existing re-sampling techniques with reduction in overall computational complexity.

In the second part of research, we examined Particle Swarm Optimization (PSO) technique to combat randomness issue in bagging which drops the performance of classifier. Ensemble model with enhanced SMOTE technique coupled with PSO selects the instances with good discriminative ability which in turn increases accuracy of the model. Proposed model result shows the improvement in micro-averaged F1-measure over existing resampling-based ensemble models on multi-class dataset.

In third part, new ensemble model is proposed to handle concept drift and imbalance joint issue for data-stream classification. To handle these issues simultaneously Hybrid Dynamic Chunk Ensemble Model has been implemented. Proposed model is chunk based ensemble model which uses the passive approach and selects the chunk size dynamically which is difficult to decide prior when data-stream is imbalanced and which has an impact on the performance of classifier. This algorithm is able to deal with dynamic Imbalance Ratio (IR) in every chunk using PSO-Ensemble. This method has evaluated on binary class and multi-class imbalanced synthetic and real-world data-streams. Evaluation results indicates consistent improvements over several existing online and chunk-based methods with different performance metrics. Proposed work can work well with data-streams with abrupt and gradual drift.

2. INTRODUCTION

2.1 Background

Today's modern multi-fold technology has become an integral part of our lives, and with the use of this technology data is also generated and continues to generate at exponential rate. Data has become valuable asset in every field. Also, intelligence have riddled in almost every sector of the world. Community is working towards data analysis and data mining algorithms to give the benefits to the society. Health/Medicine [1], Education, Retail & Banking industry [2], Tourism [3], and many different industries have started using Artificial Intelligence / Machine Learning (AI/ML) based tools to gain the profit and competitive achievement. Although the cost of storing the data and computational cost for analyzing the data is decreasing with this modern technology, complexity of analysis of data is increasing. Data used for the analysis is either available at a time which is termed as a static data or this data may be on fly which is termed as data stream. The problems with static data analysis also have similar problems with data stream analysis in addition to these some specific problems associated with it. Classification and prediction of particular instance has become vital task in data analysis. In the past most of the binary class problems were explored wherein only two distinct classes were involved. Whereas there are many real-world multi-class problems exist where more than two classes are present. When the number of classes are more than two then the complexity of multi-class classification increases. Also, in this classification underlying distribution of the instances is important. If there is an uneven distribution of data instances then there is a serious challenge of data imbalance issue. The class for which the number of instances is under-represented is called as minority class, while the rest of the class/es are called as majority class. Many traditional machine learning algorithms are biased towards the majority class which may give the undesirable results. It is expected that the model should give the unbiased result of classification for all the classes in dataset. This imbalance data issue is for static data as well as stream data. Along with imbalance issue, classification of stream data poses with concept drift issue. Concept drift is where the relationship between input variables and output is changing over the time. When concept drift and imbalance issue occur simultaneously that becomes more serious and challenging. Most of the machine learning algorithms are based on the assumption that the relationship between input and output variable is static and does not change over the time that is training and testing data are following the same distribution and this assumption fails in stream analysis.

To tackle this imbalance issue several techniques had been introduced. Ensemble learning with resampling techniques are most popular techniques, but these techniques are based on randomness. And when there is randomness then there is no guarantee that we will get always good results. Recently some intelligent techniques coupled with traditional methods have shown promising performance. But when the dataset size increases, time required by machine learning algorithms also increases in non-linear fashion. So, to deal with the time complexity distributed technologies like Hadoop, Spark has been attracted by many researchers to work with machine learning algorithms.

2.2 Research Gaps

2.2.1 Limitations in Sampling Techniques to handle imbalance issue

- Existing sampling techniques mostly designed for binary imbalanced classification problems which are not directly applicable to multi-class datasets.
- There is still scope for improvement in their efficiency in terms of execution time and effectiveness in terms of classification performance metrics for static datasets.

2.2.2 Random subset selection and machine learning in distributed environment

- Although ensemble learning with bagging is most popular research, randomness is one the drawback in subset selection of bagging algorithms.
- Random subset selection can be improved using optimized intelligent techniques.
- Machine learning in distributed environment has not explored fully. There is scope to apply machine learning in distributed environment.

2.2.3 Imbalance and Concept Drift Joint Problem in Concept Drift

- Class imbalance and concept drift can significantly affect predictive classification performance, and the problem becomes more challenging particularly when both concept drift and imbalance occur simultaneously which is not completely explored.

This research is focused on to handle imbalance issue in multi-class data ensemble learning with improved resampling technique, which is further improved by combining it with intelligent optimization technique known as Particle Swarm Optimization (PSO). Ensemble learning with PSO becomes expensive in terms of time efficiency, to make it time efficient proposed algorithm is implemented in distributed environment using Spark. Also, imbalance issue can occur in stream data, where concept drift issue distress the performance of model, concept drift issue becomes more challenging when it is combined

with imbalance issue. To address this problem hybrid dynamic chunk ensemble model is proposed.

This synopsis report begins with Chapter 2 with a review of existing challenges with the introduction and overview of overall research carried out. Chapter 3 listed objectives to fill the research gaps identified through literature survey in chapter 4. Chapter 5 consists of contribution through research while attaining the research objectives. Chapter 6 is conclusion of research. Assumptions and limitations, future scope and extension of the research which are mentioned in Chapter 7. List of publications given in Chapter 8. And references used for this research is in Chapter 9.

3. OBJECTIVES

To attain the research gaps mentioned above following research objectives are set:

1. To develop enhanced SMOTE algorithm, K-Means ApproxSMOTE (KM-ASMOTE) for binary/multiclass imbalanced dataset.
2. To develop effective and time efficient Distributed Ensemble Machine Learning model under Spark distributed environment.
3. To implement PSO based optimized resampling technique for data classification by considering discriminative ability of instances.
4. To develop novel concept drift adaptation model for the data stream analysis (incremental learning).
5. To evaluate proposed models using performance metrics and comparing them with the existing methods.

4. LITERATURE REVIEW

4.1 INTRODUCTION

Huge amount of data is generated with tremendous speed in today's era. Analysis of such massive data is a big challenge. Learning from such data using different techniques has become an important research focus. Data mining community is using different approaches to gain the insights from this data. With the advancements in technology many parallel and distributed computing techniques becoming vital to analyze such ever-growing data. But real-world data generated from different sources to analyse is not always in the desired form. It may comprise noise or it may be imbalanced in nature. Learning from such type of data is again one of the challenges before data mining community. Following sections gives the state-of-the-art work done so far on learning from such data.

4.2 Binary and Multi-class Imbalanced Problem

In the classification of data, the underlying distribution of data is uneven. The number of instances in one of the classes are under-represented than the rest of the classes, is called as imbalanced data. The under-represented class is called as minority class while the class which has large number of instances is called as majority class. It is not challenging to find the imbalance ratio or to quantify the imbalance parameter in case of binary data. Most of the traditional machine learning algorithms are biased towards this majority class which is not desirable. For example, in fraud transaction detection, fraud transactions will always be as minority class even if overall accuracy of such database is high and if minority class recall is less, it means that machine learning model has not performed well. There are two popular techniques used to deal with imbalanced dataset which are categorized as 1. Data Level, 2. Cost sensitive or algorithm level [4].

Data level techniques are modifying the underlying distribution of classes either by decreasing the number of instances from majority class known as undersampling or by increasing the number of instances from minority class known as oversampling. In second strategy that is in algorithm-level the learning algorithms are modified by considering cost of miss-classification of the instances rather than modifying underlying distribution.

This imbalanced problem becomes more difficult when data contains multiple classes as there could be multiple minority and majority classes based on certain perspectives. Existing work has not addressed such situations in data.

There are two techniques for handling multi-class imbalance, one-vs-one and one-vs-all, here multi-class problem is decomposed into binary class classification problem [5] which is handled using binary classification algorithms. But this approach has significant limitations and in one-vs-one the computing cost for decomposition grows with number of classes, dataset size.

4.2.1 Other Resampling Techniques to Handle Imbalanced Problem

The most widely used data level approach which generate synthetic instances using existing instances is SMOTE (Synthetic Minority Overssampling Technique). There are many variants of this SMOTE technique. Few of them are described below.

SMOTE: This is an oversampling technique which generates the synthetic instances using interpolation and k-nn (k-nearest neighbors) with sampling [6]. For each minority class instance this method first finds the nearest neighbors using distance metric. In the direction of all these nearest neighbour's synthetic instances are generated by finding the difference between query instance and nearest neighbors. This difference is then multiplied by random number between 0 and 1.

Borderline-SMOTE (SMOTEb): This is modified version of SMOTE which considers minority instances on the border of the minority instance region to apply SMOTE. Here based on ratio of nearest neighbor instances and minority class instances, minority instances are categorized as safe, borderline, rare and outlier. Only for the borderline instances SMOTE is applied [7].

ADAptive SYNthetic (ADASYN): This technique [8] generated instances adjacent to query minority examples that were mis-classified by a k-nearest neighbor classifier.

CCR [9]: It works in two phases, firstly, cleaning the minority class regions from majority class instances and secondly oversampling in the cleaned regions. Using this technique safe region are created without losing useful information from majority class.

MWMOTE [10]: It creates the uniform minority class regions around difficult instances by assigning a weight to each example based on its importance, and create synthetic examples.

4.2.2 Ensemble Learning

Resampling methods with ensemble learnings methods can give improved results. Ensemble learning is powerful method as every individual base learner's strength is incorporated. Bagging [11] is a popular ensemble learning technique where set of base classifiers is created using subsets of dataset. Here under-sampling and oversampling are combined with bagging techniques, called as UnderBagging and OverBagging,

respectively, and SMOTE combined with bagging is known as SMOTEBagging and these combinations has strengthened the resampling technique to balance the data.

There is another popular technique known as boosting where weights of data instances are adjusted based on their way to learn. Some examples are AdaBoost [12]. This work has been extended in [13], [14], 15] where the variants of decision tree are used. Recently ensemble learning with neural network also being explored [16]. In [17] multi-scale multi-CNN was used for the classification of skin legion. Random cropping ensemble neural (RCE-NN) [18] used for robot arm placement.

Ensemble learning with resampling techniques are further improved when these are coupled with intelligent optimization techniques. As mentioned above, in bagging random sampling of subsets is performed while boosting performs weight adjustment of difficult instances. But although both of these methods have outperformed, they are not taking into account the quality or usefulness of instances. So, the instances can be selected based on their discrimination property using optimization techniques, like PSO. Recently there are few methodologies identified in literature where intelligent techniques [19] [20] like PSO have been used for sample Search Space Optimization (SSO) [21], [22].

4.3 Concept Drift and Imbalanced Issue in Classification of Data Stream

In traditional machine learning, data to be analyzed is assumed as a static data and completely available at the time for processing. Such type of data is considered as a stationary data where underlying distribution is not changing over the time that is a concept to be learned are fixed. However, in today's digital world data is generated continuously with high volume and high speed. Some of the applications include social media analysis, Sensor network data, Web clicks or log data, telecommunication data analysis. Here continuously arriving infinite data termed as a data stream.

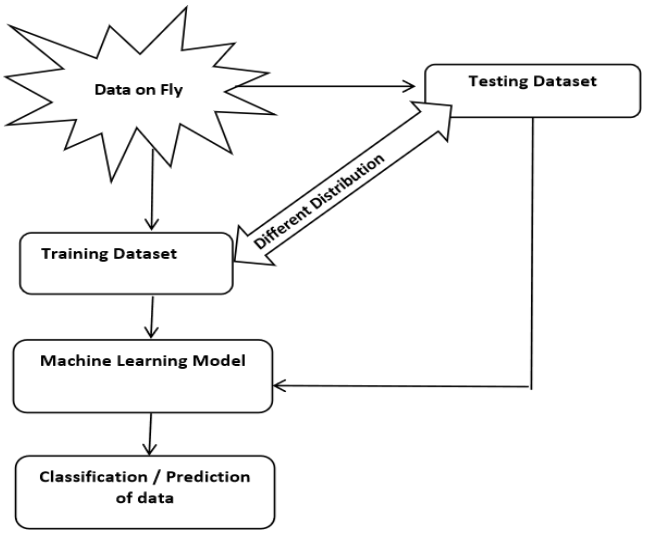


Fig. 4.1 Illustration of Concept Drift in Data Stream

In the processing of data streams, the assumptions of traditional machine learning are distressed.

Concept drift is the most common reason behind decreasing the performance of learning algorithms. Figure 4.1 depicts machine learning for data stream. Many techniques have been proposed to deal with these challenges and able to accommodate the features of data streams. These include active and passive approaches. In an active approach, explicit concept drift detectors are introduced which are raising the alarm when change is detected and then continue to process the rest of the arriving instances. This change detection is done by observing the change in the parameters like sample mean, variances and classification error of learning model [23-24]. While in passive approach, learning algorithms update themselves with the new arriving of instances which overcome the limitations of active approaches like the wrong point of change detection or no change detection. Passive approaches were implemented using single classifiers [25-26] and ensemble classifiers. Ensemble learning found as the most promising technique in data stream classification [27] than single classifiers because of its some of the merits like i) its adaptive nature to accommodate changes in the new data, ii) ability to remove aging or old irrelevant component classifiers, iii) improves accuracy by reducing variance in the error [28-29]. There are two types of ensembles i) Online Ensembles ii) Chunk based ensembles. In Online ensembles, online instance learning is used where instances in the data stream are processed by candidate classifiers one by one over time. It is a special case of a chunk-based learning where |Di| = 1 i.e. every chunk Di belonging to data stream contains a single instance. In online ensembles, online bagging and boosting [30] are very popular, while in chunk-based ensembles where most of the approaches have used fixed size chunks. Learn++ [31] NSE, AWE [32] & AUE [33] have used chunk-based ensembles.

4.4 Apache Spark Architecture

Spark [34-35] is a most effective distributed data processing engine for data intensive applications. It supports map-reduce programming paradigm and generates directed acyclic graph. It follows master-slave (worker) architecture. Master node is also known as driver node. From the driver node, a Spark Context object is created which connects to a cluster manager such as Mesos or YARN. The executor or worker performs the tasks assigned in parallel. Spark uses the Resilient Distributed Dataset (RDD) [36][37] data structure to perform the computations in parallel. The RDD stores a dataset as partitions that may be split over one or more cluster nodes. This abstraction allows the user to perform the same operations regardless how the data is distributed. Functions like map,

filter, and reduce can be run directly on RDDs and SQL-like table operations are available through the further abstracted data structures of Data Frames and Datasets.

Summary:

From the above detailed literature survey, we identified following research gaps.

- Existing sampling techniques mostly designed for mitigating the class-distribution. There is still scope for improvement in their efficiency in terms of execution time and effectiveness in terms of classification performance metrics. Machine learning in distributed environment has not explored fully. There is scope to apply machine learning in distributed environment. This issue has been addressed in Chapter 3.
- Although ensemble learning with bagging is most popular research, randomness is one the drawback in subset selection of bagging algorithms Random subset selection can be improved using optimized intelligent techniques. Chapter 4 has addressed this issue.
- Class imbalance and concept drift can significantly affect predictive classification performance, and the problem becomes more challenging particularly when both concept drift and imbalance occur simultaneously which is not completely explored. In Chapter 5 we have proposed novel approach is proposed to address this issue.

5. CONTRIBUTION

Objective I

1. To develop enhanced SMOTE algorithm, K-Means ApproxSMOTE (KM-ASMOTE) for binary/multiclass imbalanced dataset.

5.1 Background

As discussed in literature survey there are two resampling techniques to handle the imbalance problem, under-sampling and oversampling. In oversampling, SMOTE and random oversampling are the two popular approaches used. In random oversampling, existing instances in dataset are just duplicated and merged in original dataset, while in SMOTE instead of just duplicating the existing instances synthetic instances are generated which is more efficient than random oversampling technique. SMOTE algorithm is based on K-NN algorithm. To generate new synthetic instance five nearest neighbors are chosen and then average of their features is computed.

The focus of objective I is on adapting SMOTE technique to multi-class imbalanced data in a spark environment. Original SMOTE algorithm uses K-NN algorithm which is computationally expensive, as for each instance of minority class in the data set, it first calculates the Euclidean distance from the rest of all instances and then based on distance value it finds the k-nearest neighbors. Figure 5.1 depicts the how synthetic examples are generated using SMOTE. First, it chooses a random minority observation x_i, and then it randomly selects an instance x_u among the k nearest minority class neighbors of x_i. Finally, a new random sample s_i is generated by interpolating the two samples using $s_i = x_i + w \times (x_u - x_i)$, where w is a random weight in [0,1].

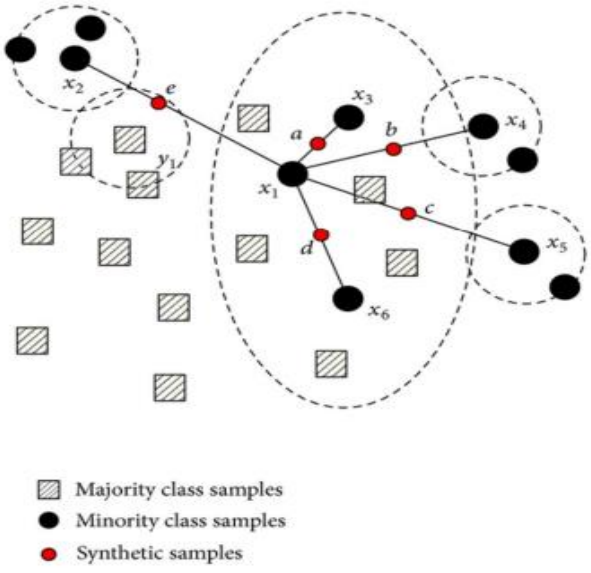


Fig. 5.1: SMOTE for generating synthetic examples (source [39])

Hence for large number of instances time complexity is high. By taking the advantage of distributed computing there were few approaches proposed in the literature to enhance this SMOTE algorithm like SMOTE-BD [40]. But again, this algorithm also has used iterative K-NN in the background. This consumes high computational power and so has limitation in the scalability. To overcome these limitations, enhanced KM-ASMOTE algorithm is proposed.

Limitations of SMOTE:

1. SMOTE synthesize new samples but it may create noisy or overlapping examples.
2. SMOTE may lead to overfitting.
3. SMOTE randomly chooses a minority instance to oversample with uniform probability.
4. It is effective to combat the imbalance issue in two class but becomes complex for multi-class imbalance as sparsely populated minority areas remain sparse.
5. It may generate the synthetic samples in the region of majority class (Fig 5.2).

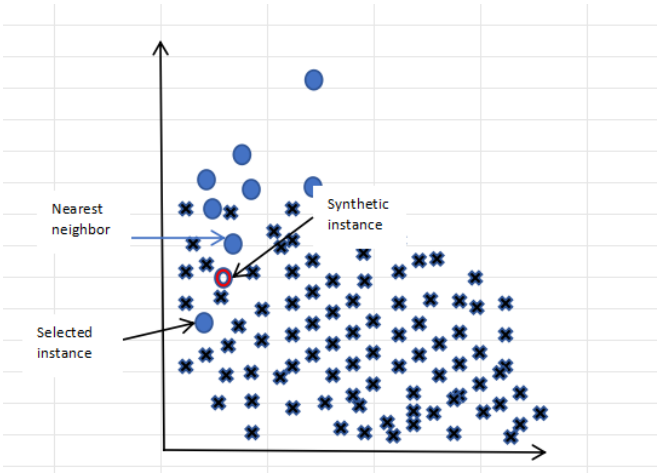
6. 

Figure 5.2: Generation of minority class samples

KM-ASMOTE

Our approach starts with generating partitions of the dataset across the worker nodes in a cluster such that each partition contains at least 50% samples from each class. Proposed method has three major steps:

1. K-means at Minority Class

In step one, majority and minority classes from the input dataset are identified. In the succeeding step, the K-means clustering algorithm is applied to all minority classes. K-means is a popular iterative method to form natural groups of data using K-cluster centroids. The K-means clustering algorithm tries to mitigate the issue of overlapping

regions due to synthetic instance generation. In K-means clustering, there are two steps involved, which are executed iteratively. In the first step, each instance is assigned to the cluster of nearest centroids. And in the second step, the updating of this cluster centroid is performed. These iterative steps are repeated until there is no movement of instances in clusters. To select the optimum number of clusters, the Elbow method is applied.

After applying the K-means clustering algorithm, the second step is the selection of clusters. In this step, clusters with minority classes are identified as shown in Fig. 5.3, and clusters containing the maximum number of minority class instances are selected, i.e., clusters that contain at least 50 % of the original minority instances. After selecting clusters, the overall number of synthetic instances to be generated are determined based on the number of majority instances, and then these are distributed among the clusters.

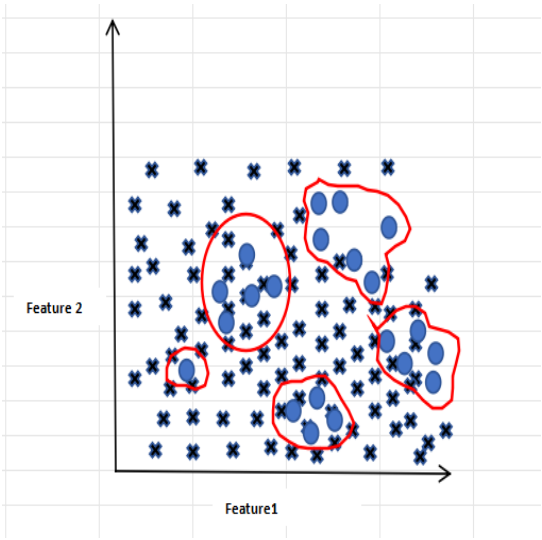


Fig. 5.3: K-means clustering at minority class

2. ASMOTE

SMOTE and SMOTE-based methods become time consuming for large class imbalance problems because these methods are based on the default K-nearest neighbour algorithm, which is a time-consuming method for balancing the minority class. If SMOTE avoids or mitigates the searching of K-nearest neighbours to generate synthetic instances, the performance of the algorithm can be improved. With this motivation, we proposed an approximate K-NN using Random Projection LSH (Locality Sensitive Hashing) i.e. RP-LSH [42-44], instead of an exact KNN search. The following paragraphs explain details of the proposed ASMOTE algorithm using LSH.

• LSH

It is hashing based algorithm to identify approximate nearest neighbors. An approximate nearest neighbor algorithm tries to reduce complexity of K-NN to sub-linear by reducing the number of comparisons needed to find the nearest neighbors. LSH works on the principle that if there are two points in feature space closer to each other, they are

very likely to have same hash value (reduced representation of data). LSH primarily differs from conventional hashing used in cryptography in the sense that cryptographic hashing tries to avoid collisions, but LSH aims to maximize collisions for similar points. Here we have used Random Projection based LSH. The subsequent section sheds light on RP-LSH.

- **Random Projection LSH**

Random projection LSH preserves certain geometric properties of high dimensional data. Here, family of hash functions is generated with the help of Random Projection. Here we have considered only sign of projection to generate hash value. Following steps are implemented for RP-LSH:

(a) Consider a dataset, D, which consists of n number of instances (x1, y1), (x2, y2),. . . . . (xn, yn).

(b) In each minority class-cluster k random vector, which is termed as a random hyperplane of length d, is created, where d is the dimension of dataset.

(c) Normalize the random vector to ensure that it has unit length.

(d) Compute the dot product of the random vector and the feature vector. If the result of the dot product is positive, assign the bit value as 1, else 0. (Refer Fig. 5.4).

(e) Concatenate all the bit values computed for k dot products.

(f) Repeat steps (c) and (d) for all instances to compute hash values for all instances.

(g) All instances with same hash value are assigned to same bucket.

Above steps are repeated for P number of iterations and through these iterations P hash tables are generated. From these hash tables all the instances of similar hash values are bucketed together by applying hamming distance. For the query instance of minority class only the instances from the bucket for which the hash values match, are considered as nearest neighbors. Synthetic samples are generated using interpolation between those nearest neighbors and the query instance.

Figure 5.5 illustrate an example of RP-LSH. In this figure h1, h2, h3 are three hyperplanes, and a, b, c, d, e, f, g are the d dimensional instances. Table 5.1 gives the hash values corresponding to those instances. These hash values are generated using Eq. 5.1

$$hash\_values_i = sign((x_p, y_p) dot (rp\_vector_k)) \quad (5.1)$$

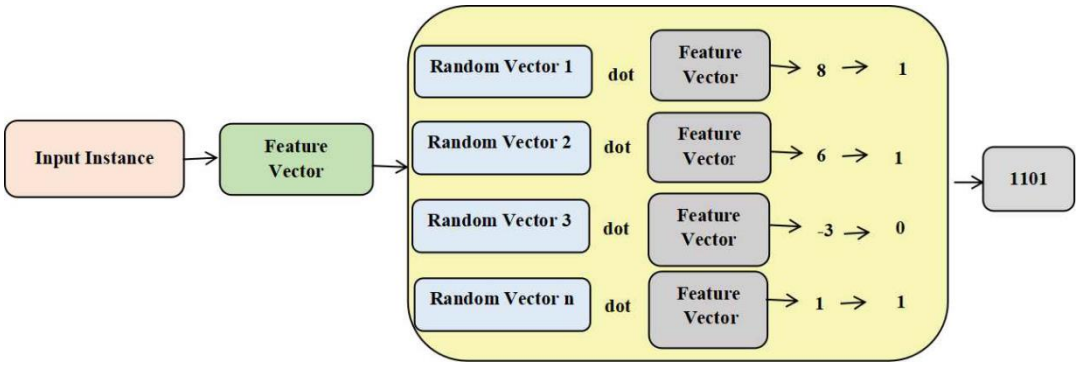


FIGURE 5.4: Hash Value Generation in RP-LSH

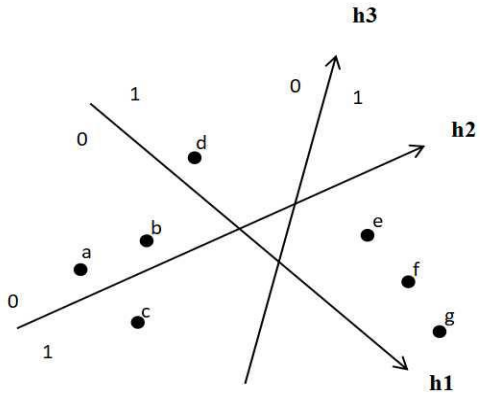


Table 5.1: Hash value for Instances.

Instances	Hash Value
ab	000
c	010
d	100
efg	111

FIGURE 5.5: Example for Random-Projection LSH

Distributed Ensemble Model Under Spark Environment:

Objective II: To develop effective and time efficient Distributed Ensemble Machine Learning model under Spark distributed environment.

After generating balanced dataset with proposed KM-ASMOTE algorithm, we have implemented distributed ensemble model using Spark environment. In ensemble model, there is combination of several classifiers which are trained on diverse set of subsets from original dataset which makes the model more powerful. To adopt Bagging in distributed environment, empirical studies where Bagging-like approach have been suggested where base learners in ensemble model are trained using disjoint partitions. But in disjoint partitions approach base learner's performance would be biased towards the majority class present in that partition. Also, if dataset is imbalanced then base classifier wouldn't be trained on all class instances from the dataset. With the motivation of recent work on distributed ensemble learning models in [35][36], we proposed Class Aware Distributed Ensemble Model (CADEM) using Spark environment.

CADEM works in following steps:

• Stratified Sampling Based Customized partition: Training dataset is distributed using customized partition across the worker nodes in Spark cluster. Here training dataset is distributed across all worker nodes based on class labels.

• Base classifiers from ensemble model are trained using distributed training dataset at each worker node using stratified cross-validation.

• Results from each worker node is accumulated at master node and using majority voting technique prediction for test dataset is performed.

Experimental Setup

This experimental analysis performed using default cluster on databricks with 1 master node and 2 to 8 worker nodes with Spark version 3.3.2.

Datasets: We have selected two multiclass imbalanced Insect and Forest Covertype datasets with continuous features. These datasets have high number of features also highly imbalanced in nature. So, these are providing good insights of different oversampling techniques. Table 5.2 describes the high-level properties of datasets used validating the proposed model.

Classifiers. We have used Random Forest Ensemble machine learning algorithm which is available in Spark.

Oversampling Techniques. We have implemented six oversampling techniques (Random Undersampling, Random Over Sampling, SMOTE, ADASYN, Borderline-SMOTE, KMeans-SMOTE) on these datasets.

Table 5.2: High-level description of datasets

Dataset Description				
Dataset	**No. of Classes**	**No. of Features**	**Feature Type**	**Imbalance ratio**
Insect Dataset	6	34	Continuous	8.65
Forest Cover Type	7	52	Continuous	103.08

Performance Metrics: Performance metrices Precision, Recall, F1-Measure are used to evaluate the existing and proposed oversampling techniques with combination of ensemble model. Here F1-measure is harmonic mean of precision and recall and weights equally.

$$Precision = \frac{TP}{TP+FP} \quad ; \quad Recall = \frac{TP}{TP+FN} \quad ; \quad F-Measure = 2 * \frac{Precision * Recall}{Precision+Recall}$$

where, TP: True Positive; FN: False Negative; FP: False Positive.

Results and discussion for Objective I & II

Proposed KM-ASMOTE algorithm is analyzed using this dataset and compared with existing sampling (Random Oversampling, Random Undersampling) and SMOTE based variants (SMOTE, SMOTEb, ADASYN) under Spark distributed environment. Fig. 5.6 and Fig. 5.7 reveals the experimental results where we have compared performance of these algorithms using F-Measure of majority and minority classes for Insect and Forest Covertype datasets, respectively.

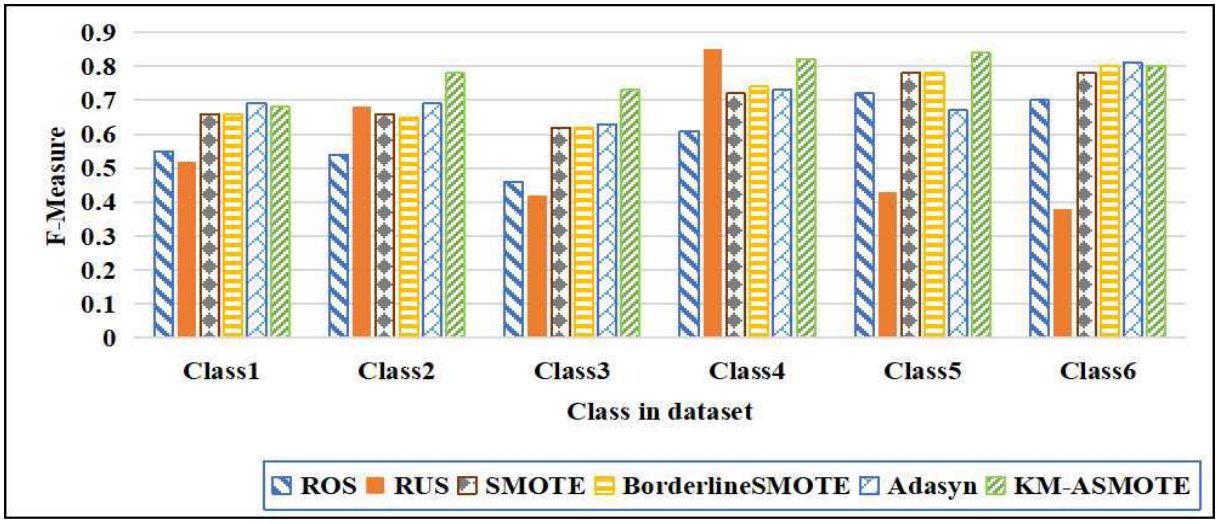


FIGURE 5.6: Comparison of different Sampling techniques using F-measure for Insect Dataset

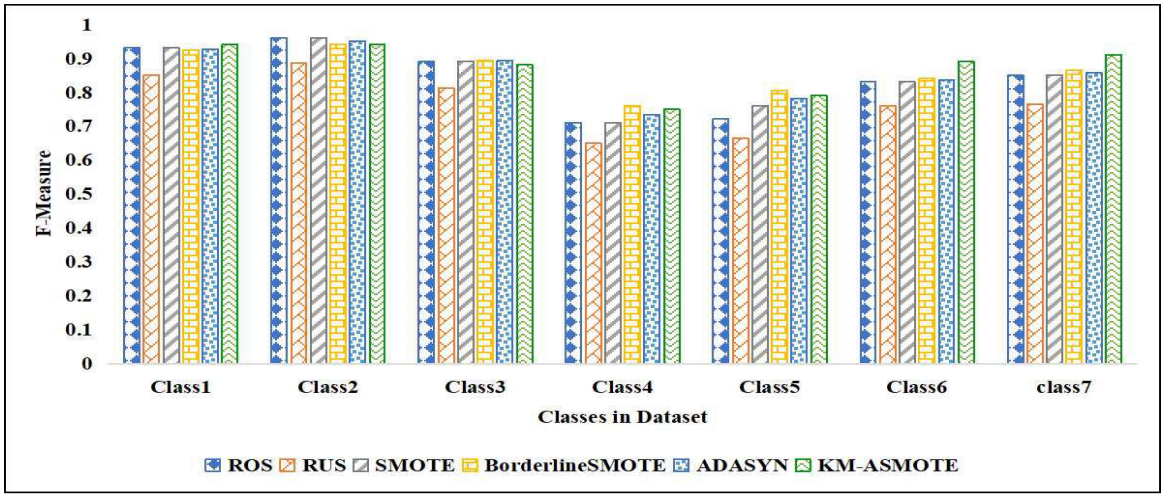


FIGURE 5.7: Comparison of different Sampling techniques using F-measure for Forest Covertype Dataset

Fig. 5.8 and Fig. 5.9 shows the performance comparison of proposed method and existing techniques at different balance ratios.

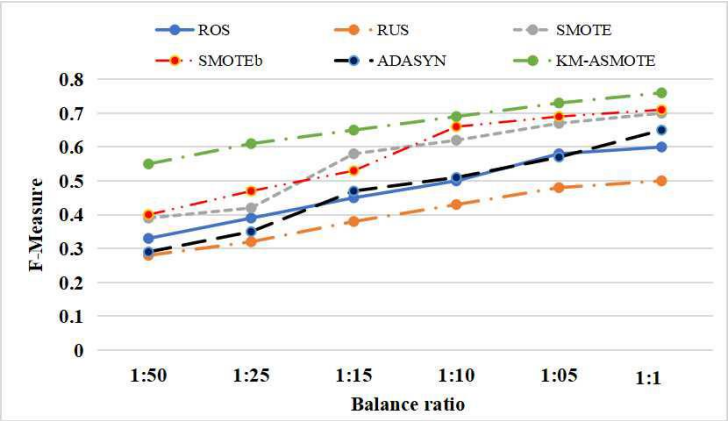


FIGURE 5.8: Performance of SMOTE variants and different Sampling techniques at different balance rations for Insect Dataset

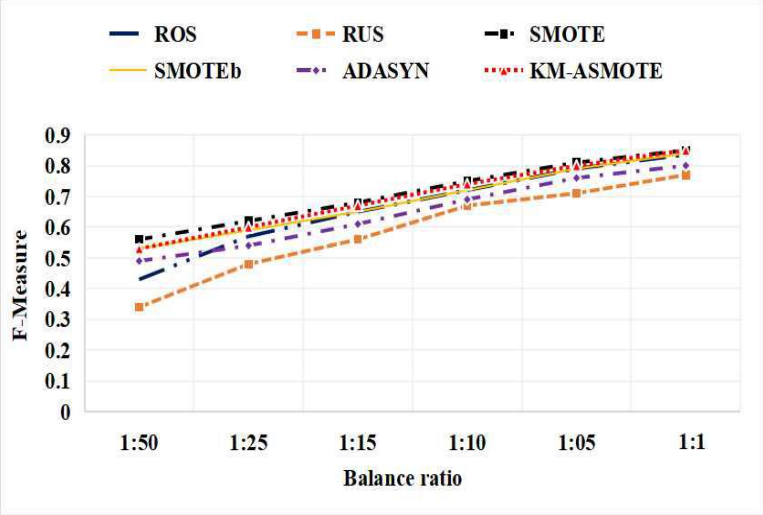


FIGURE 5.9: Performance of SMOTE variants and different Sampling techniques at different balance rations for Forest Cover Type Dataset

Naturally with increase in balance ratio all the methods have improved in their performance, however, proposed KM-ASMOTE has outperformed than existing one.

Summary:

To overcome the issue of imbalance in classification problem, SMOTE based oversampling techniques are popular. But most of the methods tend to generate noisy instances or may generate overlapping regions. Also, SMOTE method uses K-NN method while generating synthetic samples which is most expensive. To overcome this limitation, we have proposed KM-ASMOTE method which first identifies the safe areas to generate synthetic samples by applying clustering technique over minority classes. After applying clustering instead of using K-NN method to select nearest neighbor, we have proposed Random Projection based approximate nearest neighbour technique while generating synthetic samples using SMOTE. Balanced dataset is further used to train the ensemble

model under SPARK distributed environment we proposed CADEM model. Results showed that proposed approach has an improvement of 20 % in F-Measure and 18 % in minority class recall over existing SMOTE techniques with Ensemble model.

2. Objective III

To implement PSO based optimized resampling technique for data classification by considering discriminative ability of instances.

After applying KM-ASMOTE we got balanced dataset and Ensemble model trained on this data set has given improved results as compared to existing resampling techniques. However, still there is scope for improving the results because while selecting the subsets to train the model we have not considered the usefulness of the instances. So, rather than selecting all samples we can select only those samples which discriminate from the rest of the classes. So, to optimize this search space Spark based PSO ensemble model is proposed.

Proposed PSO-Bagging based Ensemble model operates in two steps:

• Step I: Balance the imbalanced dataset using oversampling technique (KM-ASMOTE) proposed under objective I.

• Step II: Find optimal subsets from balanced dataset obtained using PSO-Bagging based ensemble.

In PSO-sampling procedure, PSO is used to check the usefulness of each instance from the balanced dataset generated in Step I. To do this different sample subsets are generated which comprises instances from all classes for bagging based model formation and evaluation using test fold. From most frequently selected instances, training subsets are formed. There will be n number of iteration until desired termination criteria or till the maximum number of iterations are reached. These optimized subsets are used to train the ensemble model. Thus, the model which is trained using optimized subsets is used for evaluation and prediction of class labels of unknown instances. The model constructed here is giving unbiased evaluation, as training and testing datasets are kept completely independent.

Results and Discussion

We implemented PSO-Ensemble models like PSO-ROS, PSO-US, PSO-SMOTEb, PSOADASYN, PSO-SMOTE and PSO-KMASMOTE and all are evaluated using four datasets which are described in Section 4.3. All these models are using ensemble learning with base classifier as Logistic Regression, Decision Tree and Naive Bayes classifier. In Table 5.3, we have summarized overall performance metrics like G-mean, F-Measure and

AUC. Average accuracy is not suitable for imbalanced dataset. From Table 5.2 and literature it has been observed that oversampling techniques coupled with ensemble learning to handle imbalanced dataset have always improved the performance than the ensemble learning without any resampling technique. Moreover, proposed PSO-Ensemble models have shown significant improvement in performance in terms of F-Measure, G-Mean and Accuracy as compared to ROS, US, SMOTEb, ADASYN Ensemble models. To statistically support the experimental result, we have applied Wilkinson-Signed Rank Test on all three performance metrics.

Table 5.3: Overall performance of over-sampling techniques coupled with PSO

Dataset	Performance Metric	Without Sampling	PSO-US	PSO-ROS	PSO-SMOTE	PSO-SMOTEb	PSO-ADASYN	PSO-KMASMOTE
Wine	F-Measure	0.94	0.956	0.967	0.969	0.97	0.938	0.977
	G-mean	0.95	0.958	0.969	0.98	0.986	0.956	0.988
	AUC	0.96	0.978	0.98	0.99	0.987	0.979	0.99
Contraceptive	F-Measure	0.49	0.512	0.59	0.62	0.65	0.64	0.658
	G-mean	0.5	0.521	0.58	0.64	0.66	0.63	0.639
	AUC	0.59	0.63	0.65	0.67	0.68	0.65	0.67
Car Evaluation	F-Measure	0.72	0.74	0.77	0.79	0.81	0.78	0.767
	G-mean	0.73	0.74	0.77	0.81	0.82	0.79	0.77
	AUC	0.74	0.77	0.81	0.83	0.85	0.84	0.85
Shuttle	F-Measure	0.88	0.89	0.91	0.92	0.95	0.92	0.93
	G-mean	0.84	0.86	0.92	0.95	0.96	0.89	0.956
	AUC	0.92	0.93	0.96	0.97	0.97	0.95	0.97
Yeast	F-Measure	0.4	0.45	0.52	0.59	0.62	0.61	0.64
	G-mean	0.56	0.58	0.59	0.63	0.65	0.59	0.649
	AUC	0.64	0.67	0.71	0.74	0.76	0.709	0.75
Page Blocks	F-Measure	0.72	0.75	0.78	0.8	0.81	0.8	0.82
	G-mean	0.7288	0.78	0.79	0.81	0.81	0.82	0.81
	AUC	0.843	0.88	0.89	0.91	0.93	0.88	0.905
Thyroid	F-Measure	0.75	0.77	0.78	0.79	0.83	0.78	0.81
	G-mean	0.77	0.79	0.8	0.83	0.84	0.83	0.82
	AUC	0.81	0.85	0.87	0.88	0.89	0.86	0.88
Balance Scale	F-Measure	0.81	0.83	0.82	0.83	0.84	0.84	0.83
	G-mean	0.82	0.85	0.85	0.84	0.84	0.83	0.84
	AUC	0.88	0.89	0.91	0.93	0.94	0.95	0.94
Heart	F-Measure	0.8025	0.82	0.84	0.85	0.85	0.83	0.85
	G-mean	0.8066	0.83	0.85	0.86	0.88	0.85	0.87
	AUC	0.8566	0.87	0.88	0.91	0.91	0.902	0.92
Insect Dataset	F-Measure	0.59	0.61	0.64	0.67	0.69	0.71	0.69
	G-mean	0.601	0.6	0.65	0.66	0.68	0.7	0.67
	AUC	0.734	0.73	0.76	0.77	0.78	0.75	0.778
Forest CoverType	F-Measure	0.81	0.82	0.85	0.89	0.91	0.89	0.918
	G-mean	0.823	0.84	0.84	0.85	0.867	0.83	0.87
	AUC	0.87	0.88	0.92	0.93	0.94	0.93	0.94
FARS	F-Measure	0.537	0.55	0.59	0.61	0.65	0.62	0.64
	G-mean	0.546	0.55	0.56	0.59	0.59	0.57	0.58
	AUC	0.678	0.68	0.68	0.69	0.71	0.71	0.708
Pokerhand	F-Measure	0.373	0.39	0.45	0.49	0.53	0.56	0.55
	G-mean	0.398	0.43	0.47	0.55	0.55	0.54	0.57
	AUC	0.589	0.61	0.65	0.67	0.68	0.66	0.67
KDD'CUP99	F-Measure	0.923	0.934	0.95	0.97	0.96	0.96	0.95
	G-mean	0.934	0.945	0.95	0.97	0.97	0.95	0.96
	AUC	0.967	0.974	0.98	0.99	0.98	0.97	0.99

5.3 Objective IV: To develop novel concept drift adaption model for the data which is available on fly (data-stream).

Concept drift and imbalance joint issue lowers the performance classification model. To support this hypothesis experimental analysis of online and chunk-based algorithms on synthetic data-streams is performed using Massive Online Analysis (MOA)

tool. Reason behind selecting synthetic data-streams is that we can specify concept drift position and imbalance ratio manually.

Size of each dataset is 5,00,000. The description of these dataset is given in Table 5. As a representative, we have selected online learning models as OzaBagging, Dynamic Weight Majority (DWM) and OzaBagging's Adaptive Size Hoeffding Trees (ASHT) and chunk-based models as Adaptive Chunk Dynamic Weight Ensemble Model (ACDWEM) [41] and Adaptive Windowing Ensemble (AWE), Accuracy Updated Ensemble (AUE) which are evaluated using MOA framework. These ensemble models are tested with 25 number of base classifiers. While, Hoeffding Tree is used as a base classifier as it is a more promising classifier for data streams. Figure 5.10, 5.11, 5.12 shows how performance metrics of AWE drops at concept drift location for SEA dataset with imbalance ratios 50:50,80:20 100:1, respectively.

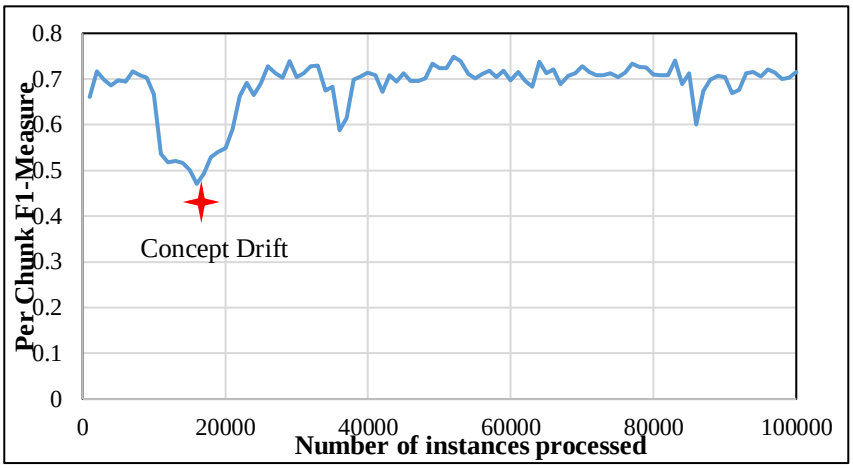


Fig. 5.10: F1-Measure drop of AWE algorithm at concept drift for SEA dataset with imbalance ratio 50:50

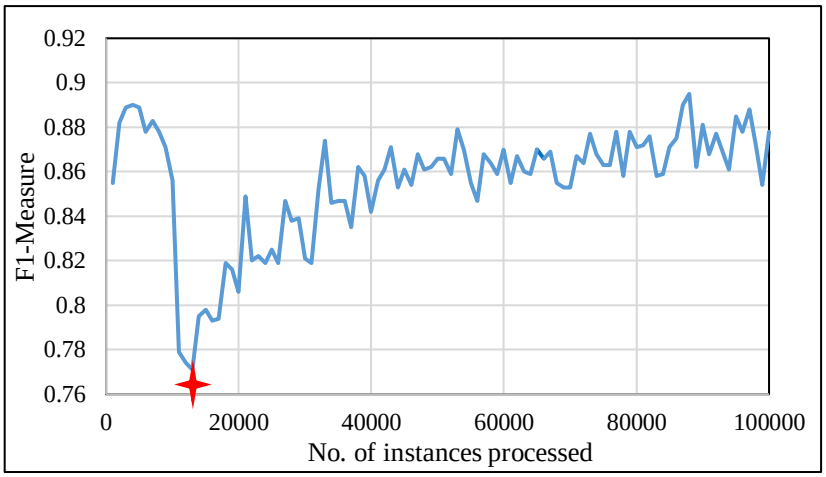


Fig. 5.11: F1-Measure drop of AWE algorithm at concept drift for SEA dataset with imbalance ratio 80:20

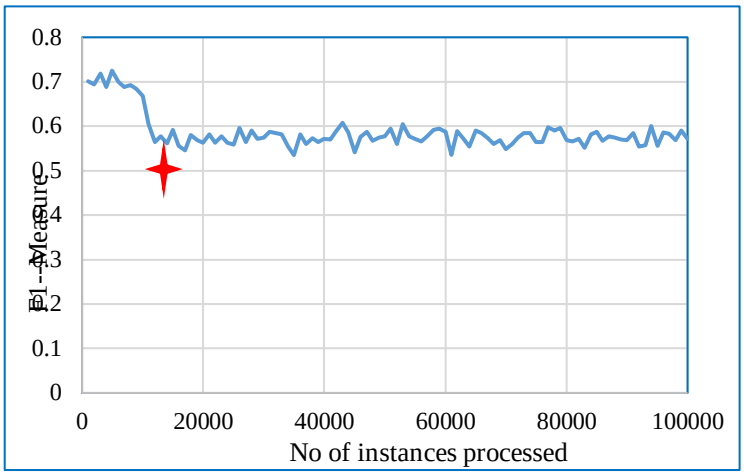


Fig. 5.12: F1-Measure drop of AWE algorithm at concept drift for SEA dataset with imbalance ratio 100:1

Conclusion from experimental analysis: Considering model cost in terms of evaluation time required and overall accuracy of classifier, chunk-based ensemble models accuracy is 5% more than online ensemble models for all datasets. All these chunks-based models assume fixed chunk size and these algorithms are computationally expensive. Objective three focuses on deciding chunk size dynamically and implementing model for data stream analysis in distributed environment.

Hybrid Dynamic Chunk Ensemble Model for Objective III

To address the concept drift and imbalance issue in data stream, novel dynamic chunk size selection algorithm is implemented.

Background:

A data stream consists of data on fly or it is an unbounded sequence of instances (e.g. Tweets, sensor data, call records) which arrive continuously with time varying intensity. In data-stream classification main challenge is underlying distribution of instances and the relationship between input features and class label is varying over the period of time known as concept drift.

Formally at time t, every instance is generated by a source with a joint probability distribution Pt (x, y). Concepts in data are stable if all instances are generated by the same distribution. But if at time t and t + Δ, there exits x such that $Pt(x, y) \neq Pt+\Delta (x, y)$, then concept drift has occurred Concept drift and imbalance issue becomes more challenging when they occur simultaneously [3]. To deal with this issue in literature there two approaches used, online and chunk based [4][5].

Ensemble models with chunk based approach is more prominent one. In chunk based approach most of the previous work used fixed-size chunks.

Limitation of fixed-size chunk ensemble model:

For imbalanced data stream if the component classifier is trained on fixed sized imbalanced data chunk may be unstable as there could be limited or zero instances from minority class. And if such component classifier is integrated in an ensemble model, model will become unstable. Fixed size chunks may not represent true distribution.

To overcome this limitation Hybrid Dynamic Chunk Ensemble Model is proposed which is able to adapt the concept drift without explicit detection and imbalance issue.

Assumption: Here assumption is data stream is available from a particular source. To evaluate proposed model insect data [47] is used. Which is optical sensor data. (Here, we are not considering real time streaming data rather we have converted static data in the form of stream on Spark architecture).

Hybrid Dynamic Chunk Ensemble Model:

In imbalanced data stream, addressing concept drift issue and adapting the model to new concepts becomes challenging. Also, in such situation if we keep fixed size chunks to train the base classifiers from ensemble model, chunk may contain only instances from majority class and it will fail to learn the minority class instances, which makes ensemble learning model unstable. To overcome this limitation and combining competent features from online and chunk based ensemble learning models, we have proposed Hybrid Dynamic Chunk Ensemble Model (HDCEM). Overall architecture diagram of HDCEM is illustrated in Fig. 13.

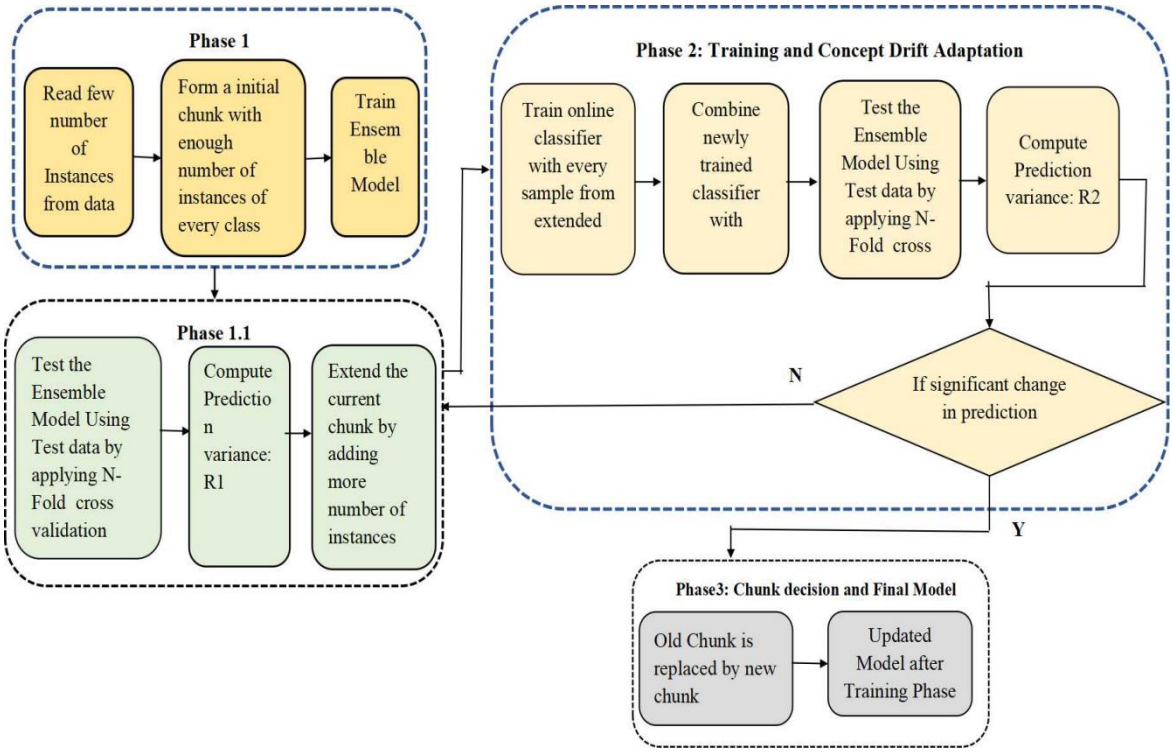

Phase 1
Read few number of Instances from data
Form a initial chunk with enough number of instances of every class
Train Ensem ble Model
Phase 1.1
Test the Ensemble Model Using Test data by applying N-Fold cross validation
Compute Predictio n variance: R1
Extend the current chunk by adding more number of instances
Phase 2: Training and Concept Drift Adaptation
Train online classifier with every sample from extended
Combine newly trained classifier with
Test the Ensemble Model Using Test data by applying N-Fold cross
Compute Prediction variance: R2
If significant change in prediction
N
Y
Phase3: Chunk decision and Final Model
Old Chunk is replaced by new chunk
Updated Model after Training Phase

Fig. 5.13: Overall architecture diagram for concept drift adaption in data stream

Detailed steps of HDCEM model are given below:

1. Chunk Formation and Training:

• At timestamp t-1 first chunk Dt is formed with enough no of samples from all minority and majority classes in the dataset.

• Find the minority and majority class count and compute imbalance ratio using Eq. 5.2 where Maj_{cnt} and Min_{cnt} is the majority class and minority class count respectively.

$$IR_t = Maj_{cnt} / Min_{cnt} \quad (5.2)$$

• If $IR_t > 1$ then imbalance in chunk is handled using optimized resampling technique. Here to balance every chunk first we have applied KM-ASMOTE algorithm.

• N base classifiers C_t…..C_{t-1} of ensemble model Ch are trained over initial chunk D_t using k-fold cross validation with PSO-Bagging and prediction variance, v_{t-1}, is calculated. Here, training of ensemble model is carried out in distributed spark environment.

• At current timestamp t previous chunk D_{t-1} is enlarged by adding more number of instances arrived forming enlarged chunk as Dt. Existing ensemble model Cht−1 is retrained using N-Fold cross validation with this chunk Dt. Also, new classifier Ct+1 is trained on every instance from enlarged chunk and combined with existing ensemble model Ct = Ct U Ct−1. After retraining the model prediction variance vt is calculated.

• To adapt with new concept and to reduce the impact of previously learned concept weight adjustment of base classifiers Ct……Ct−1 is performed. Weight assignment to each classifier in the ensemble model is done using following mechanism. Weight, w_i^t , of each base classifier depends on Mean Square Error (MSE) (Eq. 5.3) of their misclassification of instances. This weight is calculated using Eq. 5.5 on current extended chunk Dt and weight (wt_ct) of newly added base classifier is calculated using Eq. 5.5

$$MSE = \sum p(y)(1 - p(y))^2 \quad (5.3)$$

where, p(y) is probability of class for variable y. If Ch is current Ensemble model, then es is defined using Eq. 5.4

$$es = \frac{1}{|Ch|} \quad (5.4)$$

$$w_i^t = \frac{1}{es \sum p(y)(1 - p(y))^2 + MSE + \epsilon} \quad (5.5)$$

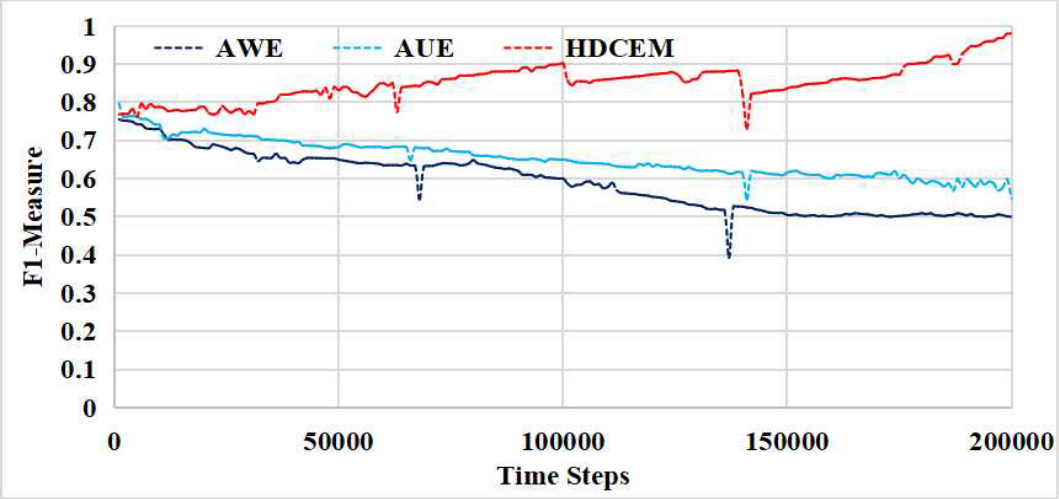
$$wt\_ct = \frac{1}{MSE + \epsilon} \quad (5.5)$$

where, ϵ is a constant which is greater than zero.

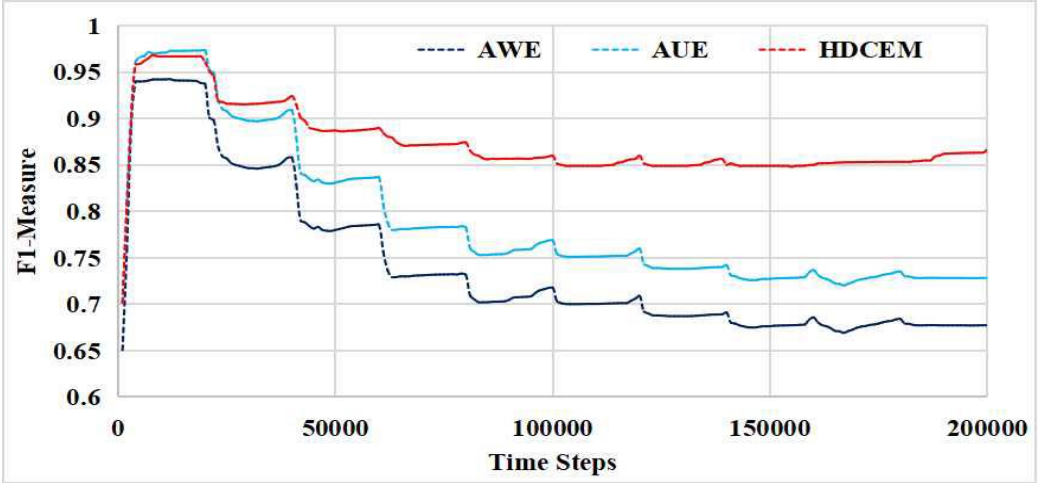
Result and Discussion for Objective III

Analysis on Different Concept Drift and Imbalance Issue

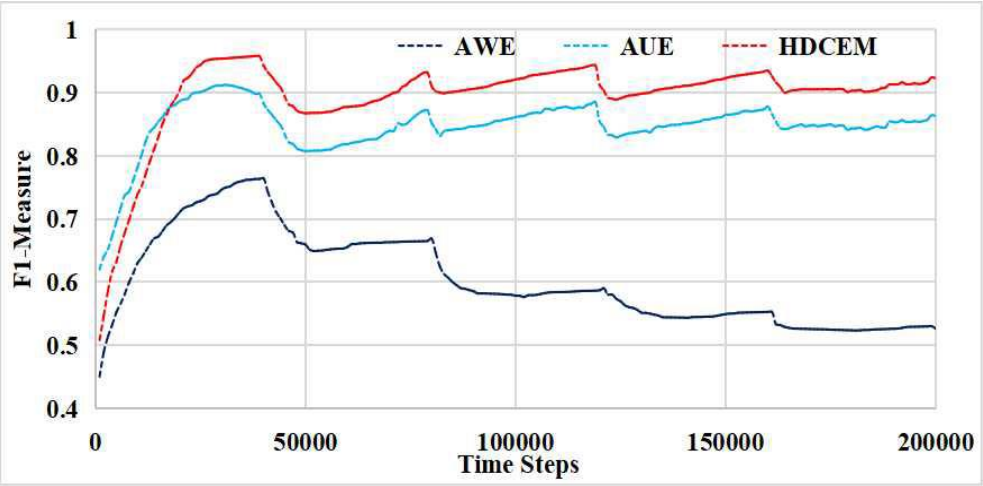
In this section we have discussed existing and proposed algorithm's performance for different concept drifts. We have generated SEA Abrupt, Hyperplane Abrupt and RandomTree Abrupt synthetic datasets with abrupt drift points at different locations by sudden change in concept. It can be observed in Fig. 5.14 that AWE, AUE algorithms dropped in F-Measure and also overall performance becomes low while proposed method has shown better performance even after these sudden abrupt drifts.



(A) SEA Abrubt Drift



(B) Hyperplane Abrupt Drift



(C) RandomTree Abrupt Drift

Fig. 5.14: Comparison of F1-measure over Abrupt Drift Dataset

The HDCEM-PSO Bagging model has following advantages:

1. It is able to perform the multi-class classification in non-stationary data streams.
2. Imbalance issue is resolved using novel PSO based resampling ensemble algorithm.
3. Combined online and chunk-based learning algorithms can handle abrupt and gradual concept drifts in imbalanced data stream.

Contribution:

1. HDCEM-PSO Bagging model is able to perform the multi-class classification in non-stationary data streams.
2. It can deal with joint issue of concept drift and imbalance in multiclass data-stream. It has used PSO based resampling ensemble (objective II) algorithm to obtain balanced dynamic sized chunks.
3. It does not store the minority class instances to handle the imbalance issue in latest chunk and able to handle dynamic imbalance ratios.
4. It uses incremental learning so no need to store past instances, and outdated or low performance component classifiers are removed using weight mechanism.
5. As the chunk size is selected dynamically its performance will not be affected due to predefined chunk size.
6. Proposed model consists of one component classifier which is trained over every instance from current chunk and all the rest of classifiers trained over overall chunk. This makes the model suitable to handle abrupt and gradual drift.

Summary

The goal of third objective is to handle imbalance and concept-drift joint issue when whole data is not available at a time. Here novel chunk based model is proposed where chunk size is selected dynamically. Chunk size has major impact on the overall prediction

accuracy specially when data is imbalanced. Proposed model not only deals with concept drift but it can also handle imbalance issue at chunk level. It produces stable ensemble model trained on high-quality chunks using optimized PSO-Bagging technique. PSOBagging adds more diversity in base classifiers and also there is control over selecting the appropriate instances. Proposed HDCEM is able to handle both gradual and abrupt drift even though chunk is imbalanced for both multi-class and binary datasets.The evaluation is conducted on both artificial datasets where concept drift is added explicitly as well as on real-world dataset. Proposed model has an improvement of 13% and 8% in F-Measure for two class data streams with abrupt and gradual drift, respectively, over existing adaptive chunk based ensemble models. HDCEM with PSO has an improvement of 20% and 21%, for multiclass data streams with abrupt and gradual concept drift, respectively.

Although proposed approach is computationally expensive on single machine, by taking the advantage of advanced distributed technologies we can balance performance and total evaluation time.

6 CONCLUSION

This chapter presents summary of the research work carried out in this thesis. This study gives a holistic view on techniques used to deal with imbalanced data issue in both static and stream data. In machine learning and data science applications, before building an application data-processing is crucial to achieve the best performance. Imbalanced dataset issue is one of the prevalent issues that needs to be handled. Existing resampling techniques like oversampling can't take the direct advantage of high-performance distributed or parallel computing techniques as their primary goal is limited to basic requirements like binary class and small dataset. Although there are numerous techniques available to deal with this issue, in this research focus is on to improve the efficiency of popular resampling technique by limiting the randomness as well as improving instance selection. When imbalance issue occurs jointly with concept drift in data stream, it reduces efficiency of the model.

SMOTE and its variants are very popular among all resampling techniques. But SMOTE techniques use K-NN algorithm during the synthetic instances and also suffered with some drawbacks like may create overlapping regions. In this study we proposed combination of k-means and approx-KNN using RP-LSH which resulted in enhanced performance metrics.

The SMOTE based resampling techniques use K-NN algorithm which makes them computationally expensive. To overcome the computational complexity, we implemented novel enhanced KM-ASMOTE Ensemble model in Spark distributed environment, wherein after applying K-Means clustering, SMOTE is implemented for every cluster using approx nearest neighbor search algorithm. We have done experimental analysis of mostly used resampling techniques and proposed KM-ASMOTE algorithm on binary and multi-class datasets using Spark distributed environment. It is found that KM-ASMOTE is 96 % faster on six worker node Spark cluster as compared to single node. KM-ASMOTE with CADEM has an improvement of 20 % in F-Measure and 18% in minority class recall over existing SMOTE techniques with Ensemble model.

We proposed another extension in KM-ASMOTE ensemble learning model in which we have applied PSO to reduce randomness in subset selection and results showed that there is further improvement in the performance. Comparison with existing resampling techniques revealed that PSO-Ensemble model in distributed environment has become more effective. PSO-KM-ASMOTE Ensemble model has an overall improvement of 29% in both F-Measure and minority class recall over existing techniques.

We have also presented our work to deal with imbalanced issue in data stream where it may occur along with concept drift. Both of these issues can be handled simultaneously using proposed Hybrid Dynamic Chunk Ensemble Model, where chunk size is decided dynamically which overcomes the limitations of fixed chunk size models. In this research wok, HDCEM produced the stable ensemble model by using dynamic chunk size selection and PSO-Bagging algorithm. Experimental results showed that proposed model is effective for the imbalanced data-streams with concept drifts. For two class data streams with abrupt concept drift, HDCEM has an average improvement of 13% and for gradual concept drift it is 8 % in F-Measure over existing chunk ensemble model. For multiclass data streams with abrupt concept drift HDCEM with PSO has an improvement of 20% and 21% improvement for data streams with gradual drift.

7. FUTURE SCOPE

This chapter offers future research directions for further investigation, possible extensions of the work outlined in the thesis, and some suggestions for expanding on previously completed work. The following problems requires an attention.

- To improve time efficiency and overall performance of oversampling techniques there is need to explore more other strategies for both multi-class and binary dataset classification.
- Here only particle swarm optimization technique is considered to optimize the search space of subsets for ensemble model.
- There is need to explore few more combinations of resampling techniques with ensemble models and other optimization techniques.
- It is required to introduce additional steps in resampling techniques which will directly address the learning difficulties in multi-class datasets.
- Current research focus is on structured data streams, there is also scope to deal with concept drift in unstructured data streams.
- Combination of different concept drift algorithms with heterogeneous ensemble models by adding more diversity can be done to gain more accuracy.
- In current work real time streaming is not used but it would be necessary to evaluate the performance of proposed model on real time streaming data like twitter tweets or any sensor data.

8. **LIST OF PUBLICATIONS**

1. Khandekar Varsha and Shrinath Pravin, “Ensemble Model for Multiclass Imbalanced Data Using Cluster Computing of Spark”, Ing´enierie des Syst`emes d’Information, Vol. 28, No. 1, pp. 161-167, ISSN: 1633-1311, February, 2023 (Scopus Indexed, Publisher: Lavoisier).

2. Varsha Khandekar and Pravin Shrinath, “Hybrid dynamic chunk ensemble model for multi-class data streams”. Indonesian Journal of Electrical Engineering and Computer Science, Volume 25, Issue 2, pp 1115–1122, February 2022. (Scopus Indexed, Publisher: Institute of Advanced Engineering and Science)

3. Varsha Khandekar and Pravin Shrinath, “A Systematic Analysis of Online and Chunk based Ensemble Models for Data Streams”, International Journal of Test Engineering and Management, Volume 83, pp 25504– 25513, Publication Issue: May - June 2020 (Scopus Indexed: Mattingley Publishing Co., Inc).

4. Varsha Khandekar and Pravin Shrinath, “Non-stationary data stream analysis: State-of-the-Art challenges and solutions”, Proceeding of International Conference on Computational Science and Applications Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-15-0790-8 8 (Publisher: Springer)

REFERENCES

[1] R. Kejriwal and Mohana, "Artificial Intelligence (AI) in Medicine and Modern Healthcare Systems," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2022, pp. 25-31, doi: 10.1109/ICAISS55157.2022.10010939.

[2] Amir M Hormozi and Stacy Giles. “Data mining: A competitive weapon for banking and retail industries”. In: Information systems management 21.2 (2004), pp. 62–71.

[3] R. Zhou, S. Lin, and M. Lin, "Design of evaluation system of residents’ tourism quality of tourism area based on fuzzy evaluation," in *Ecosystem Assessment and Fuzzy Systems Management*: Springer, 2014, pp. 189-197

[4] Nur Athirah Azhar, Muhammad Syafiq Mohd Pozi, Aniza Mohamed Din, Adam Jatowt, "An Investigation of SMOTE Based Methods for Imbalanced Datasets With Data Complexity Analysis", IEEE Transactions on Knowledge and Data Engineering, vol.35, no.7, pp.6651-6672, 2023.

[5] Alberto Fern´andez et al. “Analyzing the classification of imbalanced data sets with multiple classes: Binarization techniques and ad-hoc approaches”. In: Knowl. Based Syst. 42 (2013), pp. 97–110.

[6]. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. Smote: synthetic minority over-sampling technique. J Artif Intell Res. 2002;16:321–57.

[7] Han H, Wang W-Y, Mao B-H. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: International conference on intelligent computing. Springer; 2005. p. 878–87

[8]. He H, Bai Y, Garcia EA, Li S. Adasyn: Adaptive synthetic sampling approach for imbalanced learning.

[9] Michal Koziarski and Michal Wozniak. “CCR: A combined cleaning and resampling algorithm for imbalanced data classification”. In: *International Journal of Applied Mathematics and Computer Science* 27 (Dec. 2017), pp. 727– 736.

[10] S. Barua et al. “MWMOTE–Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning”. In: *IEEE Transactions on Knowledge and Data Engineering* 26.2 (Feb. 2014), pp. 405–425. issn: 1041-4347. doi:10.1109/TKDE.2012.232.

[11] Yoav Freund and Robert E Schapire. “A decision-theoretic generalization of on-line learning and an application to boosting”. In: European conference on computational learning theory. Springer. 1995, pp. 23–37.

[12] Zardad Khan et al. “Ensemble of optimal trees, random forest and random projection ensemble classification”. In: *Advances in Data Analysis and Classification* 14.1 (2020), pp. 97–116.

[13] Jerome H Friedman. “Greedy function approximation: a gradient boosting machine”. In: *Annals of statistics* (2001), pp. 1189–1232.

[14] Guolin Ke et al. “Lightgbm: A highly efficient gradient boosting decision tree”. In: *Advances in neural information processing systems*. 2017, pp. 3146–3154.

[15] Rodrigo Minetto, Maurıcio Pam plona Segundo, and Sudeep Sarkar. “Hydra: An ensemble of convolutional neural networks for geospatial land classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 57.9 (2019), pp. 6530–6541.

[16] Amirreza Mahbod et al. “Transfer learning using a multi-scale and multinetwork ensemble for skin lesion classification”. In: *Computer methods and programs in biomedicine* 193 (2020), p. 105475.

[17] Bo Cheng et al. “Random cropping ensemble neural network for image classification in a robotic arm grasping system”. In: *IEEE Transactions on Instrumentation and Measurement* 69.9 (2020), pp. 6795–6806

[18] Pengyi Yang, Paul D. Yoo, Juanita Fernando, Bing B. Zhou, Zili Zhang,Albert Y. Zomaya, Sample subset optimization techniques for imbalanced and ensemble learning problems in bioinformatics applications, IEEE Trans.Cybern. 44 (3) (2014) 445–455.
[19] Majdi Mafarja, Ibrahim Aljarah, Ali Asghar Heidari, Abdelaziz I. Hammouri,Hossam Faris, Al-Zoubi Ala'M, Seyedali Mirjalili, Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems, Knowl.-Based Syst. 145 (2018) 25–45.
[20] Ibrahim Aljarah, Majdi Mafarja, Ali Asghar Heidari, Hossam Faris, Yong Zhang, Seyedali Mirjalili, Asynchronous accelerating multi-leader salp chains for feature selection, Appl. Soft Comput. 71 (2018) 964–979.
[21] Seba Susan, Amitesh Kumar, SSOMaj-SMOTE-SSOMin: Three-step intelligent pruning of majority and minority samples for learning from imbalanced datasets,Applied Soft Computing, Volume 78, 2019, Pages 141-149,ISSN 1568-4946,
[22]P. Yang, P. D. Yoo, J. Fernando, B. B. Zhou, Z. Zhang and A. Y. Zomaya, "Sample Subset Optimization Techniques for Imbalanced and Ensemble Learning Problems in Bioinformatics Applications," in IEEE Transactions on Cybernetics, vol. 44, no. 3, pp. 445-455, March 2014, doi: 10.1109/TCYB.2013.2257480.
[23] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," Information Fusion, vol. 37, pp. 132-156, 2017.
[24] C. Widanage et al., "Anomaly detection over streaming data: Indy500 case study," in 2019 IEEE 12th International Conference on Cloud Computing (CLOUD), 2019, pp. 9-16: IEEE.
[25] D. Le-Phuoc, J. X. Parreira, and M. Hauswirth, "Linked stream data processing," in Reasoning Web International Summer School, 2012, pp. 245-289: Springer.
[26] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining data streams: a review," ACM Sigmod Record, vol. 34, no. 2, pp. 18-26, 2005.
[27] R. Jani, N. Bhatt, and C. Shah, "A survey on issues of data stream mining in classification," in International Conference on Information and Communication Technology for Intelligent Systems, 2017, pp. 137-143: Springer.
[28] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 2346-2363, 2018.
[29] R. H. Hariri, E. M. Fredericks, and K. M. Bowers, "Uncertainty in big data analytics: survey, opportunities, and challenges," Journal of Big Data, vol. 6, no. 1, p. 44, 2019.
[30] J. Klikowski and M. Woźniak, "Employing One-Class SVM Classifier Ensemble for Imbalanced Data Stream Classification," Cham, 2020, pp. 117-127: Springer International Publishing.
[31]Ryan Elwell and Robi Polikar. Incremental learning of concept drift in nonstationary environments. IEEE Trans. Neural Netw., 22(10):1517–1531, Oct. 2011.
[32] Haixun Wang, Wei Fan, Philip S. Yu, and Jiawei Han. Mining concept-drifting data streams using ensemble classifiers. In Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM pp. 226–235,2003.
[33] Dariusz Brzeziński and Jerzy Stefanowski. Accuracy updated ensemble for data streams with concept drift. In Hybrid Artificial Intelligent Systems. Springer, 155—163,2011.
[34] (Nov 20). Apache spark Lightning-fast unified analytics engine. Available: https://spark.apache.org/
[35] Apache Software Foundation. *RDD Programming Guide*. url: https : / / spark.apache.org/docs/latest/rdd-programming-guide.html (visited on 03/31/2019).

[36] J. Z. Huang, Y. He, C. Wei, and X. Zhang, Random sample partition data model and related technologies for big data analysis, J Data Acquisit. Process., vol. 34, no. 3, pp. 373–385, 2019
[37] J. K. Kim and Z. Wang, Sampling techniques for big data analysis, Int. Stat. Rev., vol. 87, no. S1, pp. S177–S191, 2019.

[38] V. K. Singh, M. Taram, V. Agrawal, and B. S. Baghel, A literature review on hadoop ecosystem and various techniques of big data optimization, in Advances in Data and Information Sciences, M. Kolhe, M. Trivedi, S. Tiwari, and V. Singh, eds. Singapore: Springer, 2018, pp. 231–240.

[39] Hu, F., Li, H.: A novel boundary oversampling algorithm based on neighborhood rough set model: Nrsboundary-smote. Mathematical Problems in Engineering 2013 (11 2013). https://doi.org/10.1155/2013/694809

[40] Elreedy D, Atiya AF (2019) A Comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance. Inf Sci 502:32–63.

[41] Yang Lu, Yiu-Ming Cheung, Yuan Yan Tang.,"Adaptive Chunk-Based Dynamic Weighted Majority for Imbalanced Data Streams With Concept Drift", IEEE Transactions on Neural Networks and Learning Systems, Vol 31,issue 8,pp. 2764 - 2778,2020

[42] Omid Jafari, Parth Nagarkar, and Jonathan Montaño. 2020. Improving Locality Sensitive Hashing by Efficiently Finding Projected Nearest Neighbors. In Similarity Search and Applications: 13th International Conference, SISAP 2020, Copenhagen, Denmark, September 30 – October 2, 2020, Proceedings. Springer-Verlag, Berlin, Heidelberg, 323–337. https://doi.org/10.1007/978-3-030-60936-8_25

[43]Gonzalez-Lima MD, Ludeña CC. Using Locality-Sensitive Hashing for SVM Classification of Large Data Sets. Mathematics. 2022; 10(11):1812. https://doi.org/10.3390/math10111812

[44] Nearest Neighbor Search in High Dimensions" by Piotr Indyk and Rajeev Motwani (1998)

[45]Mining of Massive Datasets" by Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman (2014)

[46] Introduction to Information Retrieval" by Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze (2008)

[47] Vinicius M. A. Souza, Gustavo E. A. P. A. Batista, Denis M. dos Reis, André G. Maletzke,"Challenges in benchmarking stream learning algorithms with real-world data",Data Mining and Knowledge Discovery 34(7):1-54,Nov 2020

[48] L. Chen, P. Chen and Z. Lin, "Artificial Intelligence in Education: A Review," in IEEE Access, vol. 8, pp. 75264-75278, 2020, doi: 10.1109/ACCESS.2020.2988510.